

本节内容

数据结构的基本概念

王道考研/CSKAOYAN.COM

学习内容

一些概念:

1.数据

数据是信息的载体,是描述客观事物属性的数、字符以及所有能够输入到计算机中并被计算机程序识别和处理的符号的集合。

2.数据元素

数据元素是**数据的基本单位**,通常作为一个整体进行考虑和处理。一个数据元素可由若干个**数据项**组成,数据项是构成数据元素的**不可分割的最小单位**。例如,学生记录就是一个数据元素,它由学号、姓名、性别等数据项组成。

3.数据类型

数据类型是一个值的集合和定义在此集合上一组操作的总称。

- 1) 原子类型:其值不可再分的数据类型。
- 2) 结构类型:其值可以再分解为若干成分(分量)的数据类型。
- 3) **抽象数据类型**:抽象数据组织和与之相关的操作。

王道考研/CSKAOYAN.COM

学习内容

这些概念大家不需要去死记硬背,随着学习深入自然会理解

4.抽象数据类型

抽象数据类型 (ADT) 是指一个**数学模型**以及定义在该模型上的一组**操作**。抽象数据类型的定义仅取决于它的一组逻辑特性,而与其在计算机内部如何表示和实现**无关**。通常用(数据对象、数据关系、基本操作集)这样的三元组来表示抽象数据类型。

PS: 数据对象

数据对象是具有相同性质的数据元素的集合,是数据的一个子集。例如,整数数据对象是集合 $N=\{0, \pm 1, \pm 2, \dots\}$ 。

5.数据结构

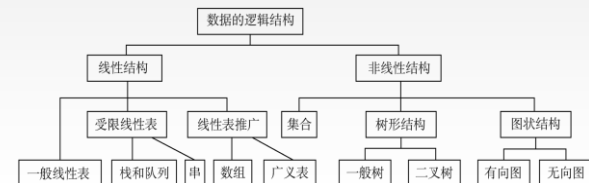
在任何问题中,数据元素都不是孤立存在的,而是在它们之间存在着某种关系,这种数据元素相互之间的**关系称为结构 (Structure)**。数据结构是相互之间存在一种或多种特定关系的数据元素的集合。数据结构包括三方面的内容:**逻辑结构、存储结构和数据的运算**。数据的逻辑结构和存储结构是密不可分的两个方面,一个算法的设计取决于所选定的逻辑结构,而算法的实现依赖于所采用的存储结构。

王道考研/CSKAOYAN.COM

逻辑结构

逻辑结构是指数据元素之间的逻辑关系,即从**逻辑关系**上描述数据。它与数据的**存储无关**,是独立于计算机的

数据的逻辑结构分为**线性结构**和**非线性结构**



王道考研/CSKAOYAN.COM

逻辑结构

集合 结构中的数据元素之间除了“同属于一个集合”的关系外，别无其他关系。类似于数学上的集合

线性结构 结构中的数据元素之间只存在**一对一**的关系。比如排队

树形结构 结构中的数据元素之间存在**一对多**的关系。比如家族族谱

图状结构或网状结构 结构中的数据元素之间存在**多对多**的关系。比如地图

王道考研/CSKAOYAN.COM

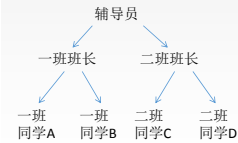
逻辑结构



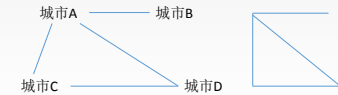
线性结构：

同学A → 同学B → 同学C → 同学D

树：通知分发



图：地图的路线分布



王道考研/CSKAOYAN.COM

物理结构

存储结构是指数据结构在计算机中的表示（又称映像），也称物理结构。它包括数据元素的表示和关

系的表示。数据的存储结构是逻辑结构用计算机语言的实现，它依赖于计算机语言。数据的存储结构

主要有：**顺序存储**、**链式存储**、**索引存储**和**散列存储**。

物理结构应该正确反映逻辑
关系
肉体-----灵魂

王道考研/CSKAOYAN.COM

物理结构

顺序存储：存储的物理位置相邻。（p.s. 物理位置即信息在计算机中的位置。）

链接存储：存储的物理位置未必相邻，通过 记录相邻元素的物理位置来找到相邻元素。

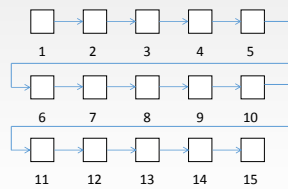
索引存储：类似于目录，以后可以联系操作系统的文件系统章节来理解。

散列存储：通过关键字直接计算出元素的物理地址（以后详解）。

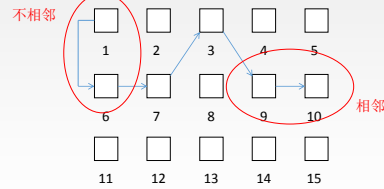
王道考研/CSKAOYAN.COM

物理结构

顺序存储：存储的物理位置相邻



链接存储：存储的物理位置**未必**相邻，通过记录相邻元素的物理位置来找到相邻元素



王道考研/CSKAOYAN.COM

数据的运算

数据的运算包括运算的定义和实现。

- 运算的定义是针对逻辑结构。
- 运算的实现是针对物理结构。

示例：

以人为例，假设有一个运算是计算人的颜值。

颜值=五官的漂亮程度之和（运算的定义）

通过读取计算机中每个人 五官的信息，然后相加得到颜值。（运算的实现）

王道考研/CSKAOYAN.COM

本节内容

算法和算法评价

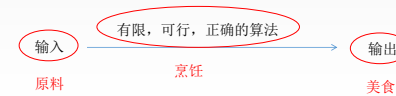
王道考研/CSKAOYAN.COM

算法的五个特征

算法是对问题求解步骤的描述，通过有限序列的指令来实现。

五大特征

- 1, 有穷性：有**有限步**之后结束
- 2, 确定性：不存在**二义性**，即没有歧义
- 3, 可行性：比如受限于计算机的计算能力，有些算法虽然理论上可行，但实际上无法完成。
- 4, 输入：能被计算机处理的各种类型数据，如数字，音频，图像等等。
- 5, 输出：一至多个程序输出结果。



王道考研/CSKAOYAN.COM

算法的时间复杂度

定义：

时间复杂度：

- 它用来衡量算法随着问题规模增大，算法执行时间**增长的快慢**；
- 是问题规模的函数： $T(n)$ 是时间规模函数 时间复杂度主要分析 $T(n)$ 的**数量级**
- $T(n)=O(f(n))$ $f(n)$ 是算法中基本运算的频度 一般我们考虑**最坏情况**下的时间复杂度

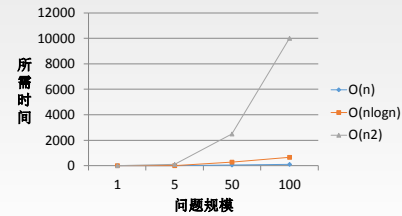
• 计算方法：取算法时间增长最快的那个函数项，忽略它的系数

空间复杂度：

- 它用来衡量算法随着问题规模增大，算法所需空间的快慢；
- 是问题规模的函数： $S(n)=O(g(n))$ ：算法所需空间的**增长率**和 $g(n)$ 的**增长率**相同。

王道考研/CSKAOYAN.COM

算法的时间复杂度



从图中，可以看到：随着问题规模的增大（横坐标），所需时间的增长率（斜率）差别很大。而且，这种差距随着**问题规模**的增大而显著地增大。

我们关心的就是算法的这个增长规模速度

常用的时间复杂度大小关系：

$O(1) < O(\log_2 n) < O(n) < O(n \log_2 n) < O(n^2) < O(n^3) < O(2^n)$ PS: $\log_2 n$ 更多写成 $\log n$

从左至右，时间性能依次降低。

王道考研/CSKAOYAN.COM



算法的时间复杂度

复杂度如何计算

考研一般会给出一段程序，然后要求计算时间复杂度。

```
int sum=0;           执行1次
for ( int i = 0; i <= n; i++){
    sum=sum+i;       int i = 0执行1次, i<=n执行n+2次, i++执行n+1次
}                   执行n+1次
```



时间分析：

该算法执行了 $3n+6$ 个语句。

假设每个语句执行时间一致，均为常数 t 。则总时间 $T = (3n+6)*t$ 。

随着问题规模 n 的增大，总时间的增长率与 n 的增长率一致，所以复杂度为 $O(n)$ 。

结论：

- 复杂度是关于**增长率**的，所以可以直接**忽视常数项**。
- 一般可以直接关注**循环体基本操作语句**（示例中的 $sum=sum+i$ ）的**执行次数**。

王道考研/CSKAOYAN.COM



算法的时间复杂度

时间复杂度计算（**单个循环体**）

直接关注循环体的执行次数，设为 k

```
int sum=0;
for ( int i = 0; i <= n; i++){
    sum=sum+i;
}
```

i 为1 执行了1次

i 为2 执行了2次

.....

i 为 n 执行了 n 次 此时仍然满足条件

$i=n+1$ 时，跳出循环

所以循环体执行次数： $k=n+1$

故时间复杂度为 $O(n)$ 。

```
int sum=0;
for ( int i = 1; i <= n; i=2*i){
    sum=sum+i;
}
```

i 取值：1,2,4,8...

满足条件： $2^k \leq n$

$k > \log_2 n$ 时，跳出循环

所以循环体执行次数： $\lceil \log_2 n \rceil$

故时间复杂度为 $O(\log n)$ 。

王道考研/CSKAOYAN.COM



算法的时间复杂度



时间复杂度计算（多个循环体）

两个运算规则：乘法规则，加法规则。

```
for (i=1; i<=n; i++)
    x++;
```

```
for (i=1; i<=n; i++)
    for (j=1; j<=n; j++)
        x++;
```

两个循环体是独立的，采用加法规则：
 $T(n)=T1(n)+T2(n)=\max(T1(n), T2(n))$
 $=O(n^2)$

```
for (int i = 1; i <= n; i++){
    for(int j=1; j<=n; j=2*j){
        sum=sum+j;
    }
}
```

两个循环体是嵌套的，采用乘法规则：
 $T(n)=T1(n)*T2(n)$
 $=O(n\log n)$

王道考研/CSKAOYAN.COM

算法的空间复杂度

空间复杂度计算

空间复杂度 $S(n)$ 指算法运行过程中所使用的辅助空间的大小。

记为：

$$S(n)=O(f(n))$$

- 辅助空间：除了存储算法本身的指令、常数、变量和输入数据外，还需要存储对数据操作的存储单元。
- 算法原地工作是指算法所需的辅助空间是常量，即 $O(1)$ 。
- 考研中出现 $O(1)$ ， $O(n)$ 较多。

王道考研/CSKAOYAN.COM